Xxxxxx

单片机原理及应用 实验报告

|  |  |
| --- | --- |
| 成绩 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 专业班级 |  | 实验日期 |  |
| 姓 名 |  | 学 号 |  |
| 实验名称 | D/A应用—信号发生器 | 指导教师 |  |

（报告内容包括实验目的、实验设备及器材、实验内容和要求、实验步骤、实验小结等）

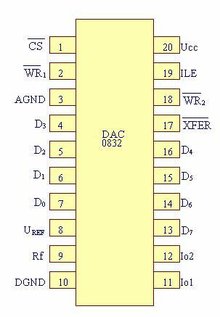
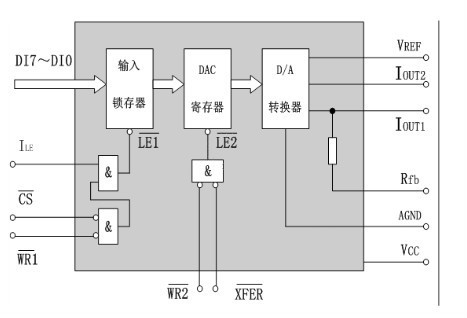
1. **实验目的**
2. 熟悉DAC0832的工作原理；
3. 熟悉 DAC0830的控制代码书写；
4. 利用单片机的D/A转换功能进行简单的设计。
5. **实验内容及原理**
   1. **DAC0832引脚图和工作原理图**

图2.DAC0832工作原理图

图1.DAC0832引脚图

* 1. **DAC0832简要介绍**

DAC0832是8位D-A转换器，它采用先进的COMS工艺制造，采用单片双列直插式封装。DAC0832有三种工作方式:**直通方式**、**单缓冲方式**和**双缓冲方式**。

**（1）单缓冲方式**。单缓冲方式是控制输入寄存器和DAC寄存器同时接收资料，或者只用输入寄存器而把DAC寄存器接成直通方式。此方式适用只有一路模拟量输出或几路模拟量异步输出的情形。

**（2）双缓冲方式。**双缓冲方式是先使输入寄存器接收资料，再控制输入寄存器的输出资料到DAC寄存器，即分两次锁存输入资料。此方式适用于多个D/A转换同步输出。

**（3）直通方式。**直通方式是资料不经两级锁存器锁存，即 CS\*，XFER\* ，WR1\* ，WR2\* 均接地，ILE接高电平。此方式适用于连续反馈控制线路和不带微机的控制系统，不过在使用时，必须通过另加I/O接口与CPU连接，以匹配CPU与D/A转换。

知识拓展（了解不用抄）

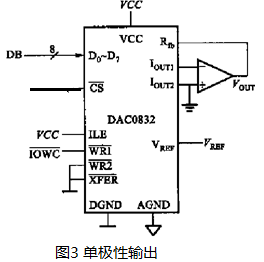
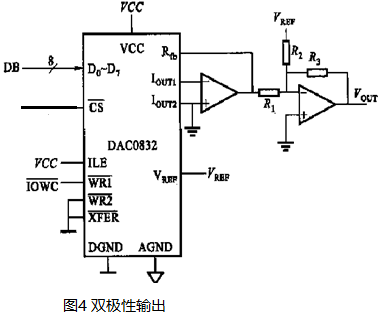
①单极性输出

[单极性电压输出电路](https://baike.so.com/gallery/list?ghid=first&pic_idx=7&eid=5414862&sid=5653004" \t "_blank)如图3所示, 由运算放大器进行电流→电压转换，使用内部反馈电阻。输出电压值VOUT和输入数字量D的关系:

VOUT = - VREF ×D/256

D = 0~255， VOUT = 0 ~ - VREF ×255/256

VREF = -5V， VOUT =0~5×(255/256)V

VREF = +5V， VOUT = 0 ~ -5×(255/256)V

②双极性输出

如果实际应用系统中要求输出模拟电压为双极性，则需要用转换电路实现。如右图4所示[双极性电压输出电路](https://p1.ssl.qhimg.com/t01a16a6c840c09aebf.png).

其中 R2=R3=2R1

VOUT= 2×VREF×D/256 -VREF= (2D/256-1)VREF

D = 0时， VOUT= -VREF;

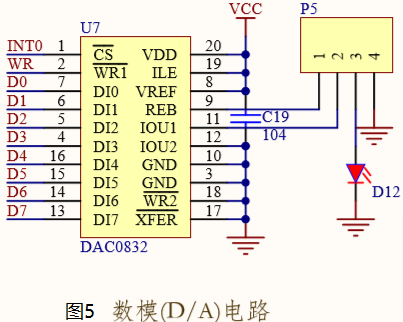
D = 128时， VOUT= 0;

D = 255时，

VOUT= (2×255/256-1)×VREF= (127/128)VREF

即:输入数字为0~255时，输出电压在- VREF ~+ VREF之间变化。

1. **开发板实验电路**



开发板DAC0832电路如图5，图中已连接WR2、XFER到地，已连接ILE到高电平，因此该电路为单缓冲方式，控制信号仅为一级缓冲控制信号CS和WR1。DAC08432为电流输出型器件，可将P5引脚用短接片短接后，使电流流过发光二极管 D12，可通过发光二极管D12的发光强度观察输出电流的变化。

实际应用中，应外接运放将其输出变为单极性或双极性电压输出型，见图3、图4。

1. **实验要求**

1、读懂实验硬件电路图，熟悉DAC0832工作原理

1. 编写程序实现以下功能：

运用STM89C52和DAC0832产生正弦波，锯齿波，三角波；

当按下并弹起一个按键时，输出正弦波，第二次按下并弹起该键输出锯齿波，第三次按下并弹起该键输出三角波。如此，依次循环切换；；

1. 调试，用嵌入式逻辑分析仪观察波形是否正确；
2. 编译并下载程序，用示波器观察DAC0832输出，验证程序执行结果。

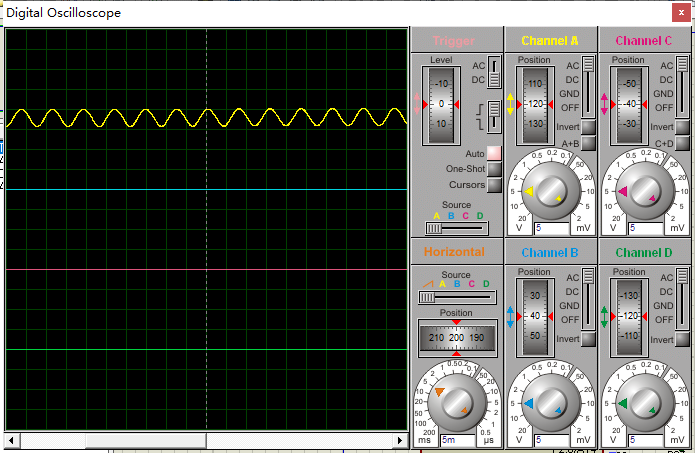
**四、实验步骤（参考）**

* 1. 实验电路图单极性输出如图3，双极性输出如图4；
  2. 上机实验，设计流程图；
  3. 编写程序建立不同波形的子函数；
  4. 书写主函数，初始化；
  5. 调用子函数；

**五、实验结果**

（拍摄不同波形的示波器显示图片并张贴在下方，抄写程序代码）

1. 示波器显示图片



1. 程序源码

// @Author  : MrSkYe

// @Email   : skye231@foxmail.com

// @File    : DA.c

// @encoding: utf-8

#include<reg51.h>

// 正弦信号发生数组

unsigned **int** code sin\_sign\_data[360]=

{0x80,0x82,0x84,0x86,0x88,0x8b,0x8d,0x8f,0x91,0x94,0x96,0x98,0x9a,0x9c,0x9e,0xa1,0xa3,0xa5,0xa7,0xa9,0xab,0xad,0xaf,0xb1,

0xb4,0xb6,0xb8,0xba,0xbc,0xbe,0xbf,0xc1,0xc3,0xc5,0xc7,0xc9,0xcb,0xcc,0xce,0xd0,0xd2,0xd3,0xd5,0xd7,0xd8,0xda,0xdc,0xdd,

0xdf,0xe0,0xe2,0xe3,0xe4,0xe6,0xe7,0xe8,0xea,0xeb,0xec,0xed,0xee,0xef,0xf0,0xf2,0xf3,0xf3,0xf4,0xf5,0xf6,0xf7,0xf8,0xf9,

0xf9,0xfa,0xfb,0xfb,0xfc,0xfc,0xfd,0xfd,0xfe,0xfe,0xfe,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,0xff,

0xff,0xff,0xfe,0xfe,0xfe,0xfd,0xfd,0xfc,0xfc,0xfb,0xfb,0xfa,0xf9,0xf9,0xf8,0xf7,0xf6,0xf5,0xf4,0xf4,0xf3,0xf2,0xf1,0xf0,

0xee,0xed,0xec,0xeb,0xea,0xe8,0xe7,0xe6,0xe4,0xe3,0xe2,0xe0,0xdf,0xdd,0xdc,0xda,0xd9,0xd7,0xd5,0xd4,0xd2,0xd0,0xce,0xcd,

0xcb,0xc9,0xc7,0xc5,0xc3,0xc2,0xc0,0xbe,0xbc,0xba,0xb8,0xb6,0xb4,0xb2,0xb0,0xae,0xab,0xa9,0xa7,0xa5,0xa3,0xa1,0x9f,0x9c,

0x9a,0x98,0x96,0x94,0x92,0x8f,0x8d,0x8b,0x89,0x86,0x84,0x82,0x80,0x7d,0x7b,0x79,0x77,0x75,0x72,0x70,0x6e,0x6c,0x69,0x67,

0x65,0x63,0x61,0x5f,0x5c,0x5a,0x58,0x56,0x54,0x52,0x50,0x4e,0x4c,0x4a,0x48,0x46,0x44,0x42,0x40,0x3e,0x3c,0x3a,0x38,0x36,

0x34,0x33,0x31,0x2f,0x2d,0x2c,0x2a,0x28,0x27,0x25,0x24,0x22,0x21,0x1f,0x1e,0x1c,0x1b,0x19,0x18,0x17,0x16,0x14,0x13,0x12,

0x11,0x10,0x0f,0x0e,0x0d,0x0c,0x0b,0x0a,0x09,0x08,0x07,0x07,0x06,0x5,0x5,0x4,0x3,0x3,0x2,0x2,0x1,0x1,0x1,0x0,0x0,0x0,0x0,

0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x1,0x1,0x1,0x2,0x2,0x3,0x3,0x4,0x4,0x5,0x6,0x6,0x7,0x8,0x9,0xa,0xa,0xb,0xc,

0xd,0xe,0xf,0x10,0x12,0x13,0x14,0x15,0x16,0x18,0x19,0x1a,0x1c,0x1d,0x1f,0x20,0x22,0x23,0x25,0x26,0x28,0x2a,0x2b,0x2d,0x2f,

0x30,0x32,0x34,0x36,0x38,0x39,0x3b,0x3d,0x3f,0x41,0x43,0x45,0x47,0x49,0x4b,0x4d,0x4f,0x51,0x53,0x55,0x58,0x5a,0x5c,0x5e,

0x60,0x62,0x65,0x67,0x69,0x6b,0x6d,0x70,0x72,0x74,0x76,0x78,0x7b,0x7d};

**void** main()

{

    unsigned **int** i=0;

**while**(1)

    {

**for**(i=0;i<360;i++)

        {

            P2=sin\_sign\_data[i];//P2输出信号

            ;;;//短时间延迟，不需要delay介入

        }

    }

}